Forecasting Research

Forecasting Research Division
Technical Report No. 126

ITERATIVE SOLUTIONS OF THE 3D MONGE-AMPERE EQUATION

by

J.G.Bonekamp

October 1994

Meteorological Office
London Road
Bracknell
Berkshire
RG12 2SZ
United Kingdom

Forecasting Research Division
Technical Report No. 126

ITERATIVE SOLUTIONS OF THE 3D MONGE-AMPERE EQUATION

PV inversion of model anomalies

J.G. Bonekamp

October 1994

© Crown Copyright 1994.

**Forecasting Research,
Meteorological Office,
London road,
Bracknell, RG 12 2SZ,
United Kingdom.**

Foreword

The aim of this report is to establish a finite difference method solving the 3D Monge-Ampère equation. The discrete nonlinear system solver considered is a Successive Overrelaxation (SOR) method improved by multigrid techniques. It extends the method as described by Fulton [2] and in principle, it is applicable to the Monge-Ampère equation of any dimension. In this report only the 3D equation is studied with respect to the applications in semi-geostrophic theory.

Results on the Dirichlet boundary value problem show the second order accuracy of the numerical method. For finer grids, the multigrid technique reduces dramatically the number of iterations needed to reach a converged state. This makes this nonlinear approach competitive with solution methods solving a leading order approximation of the Monge-Ampère equation (Poisson problem). Neumann and mixed boundary conditions are also considered. In these cases some boundary value problems are only well posed under regularity conditions and the second order accuracy is complicated by the introduction of fictituous boundary points. These boundary conditions are still under investigation, although good results are available, especially in cases consisting of both Dirichlet and other boundary conditions.

It is assumed that convex solutions of the Monge-Ampère equation exist for sufficient smooth convex boundaries. Convex solutions are used for semi-geostrophic Potential Vorticity (PV) inversion. In other words, the approximated Monge-Ampère solutions allow us to recover wind and temperature fields from PV distributions. This is shown for PV anomalies with a limited effect at the boundaries. However the appropriate problem to solve in semi-geostrophic theory is stated in the dual space, see for example Roulstone and Norbury [6]. This is still a challenge for the future.

The report starts with an introduction to the Monge-Ampère equation. In Chapter 2 a detailed description is given of the finite differences, the successive overrelaxation and the multigrid techniques. Results are shown for some exact known test functions. Chapter 3 gives an elementary introduction in the use of PV and semi-geostrophic dynamics. It continues with the description of the modelled PV anomalies followed by contour and arrow plots of the associated wind and temperature fields. At the end of the report the FORTRAN 77 subroutines are explained such that future user applications and modifications can be made without too many complications.

Hans Bonekamp

Contents

1 Introduction **3**

1.1 Definition of Monge-Ampère equation 3

1.2 First order approximation. 4

2 The numerical method **5**

2.1 Finite Differences 5

2.2 Successive Overrelaxation 7

2.3 Convex approximations 8

2.4 Boundary conditions 9

2.5 Overrelaxation and sweeping 11

2.6 Multigrid technique 11

2.7 Test function results 12

3 PV inversion in semi-geostrophic theory **17**

3.1 PV anomalies 19

3.2 Axi-symmetric cases. 19

3.3 A non-symmetric case. 23

4 The FORTRAN 77 implementation **27**

4.1 Program outline 27

4.2 Data structure 28

4.3 Procedures 30

4.4 Input/output 33

4.5 PV-Wave 33

Chapter 1

Introduction

The Monge-Ampère equation is a nonlinear partial differential equation which arises in a natural way in the consideration of various geometric problems and has therefore served as the subject of investigation of many geometers. A large body of theory exists concerning the existence and uniqueness of solutions (see e.g. the work of Pogorelov). Most of this work concentrate on the two dimensional case. However, numerical studies of the Monge-Ampère equation are a relatively recent development encouraged by the applications in rearrangement theory and semi-geostrophic dynamics .

1.1 Definition of Monge-Ampère equation

Let Ω be a bounded domain in $\mathbb{R}^n$ and consider functions $u \in C^2(\Omega) \cap C(\overline{\Omega})$, The n-dimensional Monge-Ampère operator is defined by

$$M_n[u] = \det(H(u)), \tag{1.1}$$

where $H(u)$ is the Hessian matrix of u. In this report, we consider only the Monge-Ampère equations in three dimensions

$$M_3[u] = \begin{vmatrix} u_{xx} & u_{xy} & u_{xz} \\ u_{xy} & u_{yy} & u_{yz} \\ u_{xz} & u_{yz} & u_{zz} \end{vmatrix} =$$

$$u_{xx}u_{yy}u_{zz} - u_{xx}u_{yz}^2 - u_{yy}u_{xz}^2 - u_{zz}u_{xy}^2 + 2u_{xy}u_{xz}u_{yz} = f(x,y,z). \tag{1.2}$$

A further restriction is made to allow only convex solutions,

$$u(\lambda\mathbf{x} + (1-\lambda)\mathbf{y}) \leq \lambda u(\mathbf{x}) + (1-\lambda)u(\mathbf{y}), \qquad \lambda \in (0,1), \quad \mathbf{x},\mathbf{y}, \in \overline{\Omega}, \tag{1.3}$$

because these are associated with the stable solutions of the applications in semi-geostrophic theory (see Cullen, Norbury, Purser and Shutts [1] and chapter 3). This

implies that the right-hand-side $f(x,y,z)$ has to be positive. The boundary conditions considered are of the Dirichlet, Neumann or mixed type. Some boundary value problems (e.g. Neumann boundary value problems) require additional regularity conditions to be well posed. (See next section)

Remark: In more general description of Monge-Ampère equations the right-hand-side might be a function of u and its first and second order derivatives.

1.2 First order approximation.

Consider the Dirichlet boundary value problem

$$\begin{cases} M_3[u] &= 1+\epsilon h(x,y,z), & \mathbf{x}\in\Omega, 0<|\epsilon|<<1 \\ u &= g(x,y,z), & \mathbf{x}\in\partial\Omega, \end{cases} \tag{1.4}$$

Obviously $u_0(x,y,z)=(x^2+y^2+z^2)/2$ satifies $M_3[u_0]=1$. Defining $v=(u-u_0)/\epsilon$, we can expand the Monge-Ampère operator

$$\begin{aligned} M_3[u] = \; 1+ \; & \epsilon\Delta v+\epsilon^2\left(v_{xx}v_{yy}-v_{xy}^2+v_{xx}v_{zz}-v_{xz}^2+v_{yy}v_{zz}-v_{yz}^2\right)+ \\ & \epsilon^3 M_3[v], \qquad \mathbf{x}\in\Omega, \end{aligned}$$

and a first order approximation for the solution of the Monge-Ampère problem (1.4) can be given solving the Poisson boundary value problem

$$\begin{aligned} \Delta v &= h(x,y,z), & \mathbf{x}\in\Omega, \\ v &= [g(x,y,z)-u_0(x,y,z)]/\epsilon, & \mathbf{x}\in\partial\Omega, \end{aligned}$$

Of course the same leading order technique can be applied for a different u_0 or other boundary conditions. Solution methods for Poisson boundary value problems are well known and efficient. Most of them (e.g. a Gauss-Seidel iteration) can be performed on the same grid as the proposed nonlinear method of the next chapter. Therefore, these leading order problems form a good test for the Monge-Ampère boundary value problems. In the case of solely Neumann boundary value problems we obtain solely Neumann boundary condition for the Poisson equation. The regularity condition for the Poisson boundary problem, derived from Greens theorem

$$\int_\Omega \Delta v \; dv = \int_\Omega h \; dv = \int_{\partial\Omega} \frac{\partial v}{\partial n} \; ds.$$

implies that we have regularity conditions for the associated Monge-Ampère boundary value problem as well.

Chapter 2

The numerical method

The second order differential operators present in the Monge-Ampère equation are approximated by difference operators resulting in a set of discrete equations. The discrete system is solved by a SOR (Successive Over-Relaxation) method with a modification due to the nonlinearity. This relaxation method is incorporated in a Full Approximation multigrid Scheme (FAS) to reduce the number of iterations required to reach a converged stage. The description of the method will restrict itself to the Monge-Ampère equation on a 3D block area.

2.1 Finite Differences

We introduce a 3D block $[x_0, x_1] \times [y_0, y_1] \times [z_0, z_1]$ and assume that the uniform grid sizes in x, y and z directions are, respectively, $h_x = 1/N_x$, $h_y = 1/N_y$ and $h_z = 1/N_z$. The grid points are indicated by

$$(x_i, y_j, z_k) = (x_0 + ih_x, y_0 + jh_y, z_0 + kh_z),$$

with $i = 0, .., N_x$, $j = 0, .., N_y$ and $k = 0, .., N_z$. $u_{i,j,k}$, $U_{i,j,k}$ and $f_{i,j,k}$ will define respectively the approximated solution values, exact solution values and values of the right hand side in the grid point (x_i, y_j, z_k). Thus,

$$\begin{aligned} u_{i,j,k} &\approx u(x_i, y_j, z_k) \\ U_{i,j,k} &= u(x_i, y_j, z_k) \\ f_{i,j,k} &= f(x_i, y_j, z_k) \end{aligned}$$

The second order centered finite differences approximations for the relevant derivatives are given by

$$
\begin{aligned}
u_{x;i,j,k} &= [u_{i+1,j,k} - u_{i-1,j,k}]/2h_x \\
u_{y;i,j,k} &= [u_{i,j+1,k} - u_{i,j-1,k}]/2h_y \\
u_{z;i,j,k} &= [u_{i,j,k+1} - u_{i,j,k-1}]/2h_z
\end{aligned}
$$

$$
\begin{aligned}
u_{xx;i,j,k} &= [u_{i+1,j,k} + u_{i-1,j,k} - 2u_{i,j,k}]/h_x^2 \\
u_{yy;i,j,k} &= [u_{i,j+1,k} + u_{i,j-1,k} - 2u_{i,j,k}]/h_y^2 \\
u_{zz;i,j,k} &= [u_{i,j,k+1} + u_{i,j,k-1} - 2u_{i,j,k}]/h_z^2
\end{aligned}
$$

$$
\begin{aligned}
u_{xy;i,j,k} &= [u_{i+1,j+1,k} + u_{i-1,j-1,k} - u_{i+1,j-1,k} - u_{i-1,j+1,k}]/4h_xh_y \\
u_{xz;i,j,k} &= [u_{i+1,j,k+1} + u_{i-1,j,k-1} - u_{i+1,j,k-1} - u_{i-1,j,k+1}]/4h_xh_z \\
u_{yz;i,j,k} &= [u_{i,j+1,k+1} + u_{i,j-1,k-1} - u_{i,j+1,k-1} - u_{i,j-1,k+1}]/4h_yh_z
\end{aligned}
$$

The substitution of these approximations yields the following second order approximation for the Monge-Ampère equation in grid point (x_i, y_j, z_k):

$$M^h[u_{i,j,k}] = f_{i,j,k}, \tag{2.1}$$

where the discrete 3D Monge-Ampère operator M^h is defined by

$$M^h[u_{i,j,k}] = \begin{vmatrix} u_{xx;i,j,k} & u_{xy;i,j,k} & u_{xz;i,j,k} \\ u_{xy;i,j,k} & u_{yy;i,j,k} & u_{yz;i,j,k} \\ u_{xz;i,j,k} & u_{yz;i,j,k} & u_{zz;i,j,k} \end{vmatrix}. \tag{2.2}$$

Remark: Other one- or more step finite differences are possible and may appear to be preferable in some applications. We give here a few other one-step possibilities. The cross derivative finite differences can be replaced by the general second order differences,

$$
\begin{aligned}
u_{xy;i,j,k} &= [\alpha \quad (u_{i+1,j+1,k} - u_{i+1,j,k} - u_{i-1,j,k} \\
&\quad +u_{i-1,j-1,k} - u_{i,j+1,k} - u_{i,j-1,k} + 2u_{i,j,k})+ \\
&\quad (1-\alpha) \quad (-u_{i+1,j-1,k} + u_{i+1,j,k} + u_{i-1,j,k} \\
&\quad -u_{i-1,j+1,k} + u_{i,j+1,k} + u_{i,j-1,k} - 2u_{i,j,k})]/2h_xh_y, \\
u_{xz;i,j,k} &= [\beta \quad (u_{i+1,j,k+1} - u_{i+1,j,k} - u_{i-1,j,k} \\
&\quad +u_{i-1,j,k-1} - u_{i,j,k+1} - u_{i,j,k-1} + 2u_{i,j,k})+ \\
&\quad (1-\beta) \quad (-u_{i+1,j,k-1} + u_{i+1,j,k} + u_{i-1,j,k} \\
&\quad -u_{i-1,j,k+1} + u_{i,j,k+1} + u_{i,j,k-1} - 2u_{i,j,k})]/2h_xh_z, \\
u_{yz;i,j,k} &= [\gamma \quad (u_{i,j+1,k+1} - u_{i,j+1,k} - u_{i,j-1,k} \\
&\quad +u_{i,j-1,k-1} - u_{i,j,k+1} - u_{i,j,k-1} + 2u_{i,j,k})+ \\
&\quad (1-\gamma) \quad (-u_{i,j+1,k-1} + u_{i,j+1,k} + u_{i,j-1,k} \\
&\quad -u_{i,j-1,k+1} + u_{i,j,k+1} + u_{i,j,k-1} - 2u_{i,j,k})]/2h_yh_z,
\end{aligned}
$$

with $\alpha, \beta, \gamma \in [0,1]$. Considering a cubic grid ($h = h_x = h_y = h_z$), we can replace the homogeneous difference operators by the following second order differences, which could make the discrete system more sparse.

$$\begin{aligned} u_{xx;i,j,k} &= [\ \ A + B - C - 4u_{i,j,k}]/4h^2, \\ u_{yy:i,j,k} &= [\ \ A - B + C - 4u_{i,j,k}]/4h^2, \\ u_{zz:i,j,k} &= [-A + B + C - 4u_{i,j,k}]/4h^2, \end{aligned} \tag{2.3}$$

where

$$\begin{aligned} A &= u_{i+1,j+1,k} + u_{i-1,j-1,k} + u_{i+1,j-1,k} + u_{i-1,j+1,k}, \\ B &= u_{i+1,j,k+1} + u_{i-1,j,k-1} + u_{i+1,j,k-1} + u_{i-1,j,k+1}, \\ C &= u_{i,j+1,k+1} + u_{i,j-1,k-1} + u_{i,j+1,k-1} + u_{i,j-1,k+1}. \end{aligned}$$

2.2 Successive Overrelaxation

The Jacobi iteration is defined by a successive displacement of the approximation $u_{i,j,k}$ in all the grid points. Assume $u^m_{i,j,k}$ is the approximation in the grid point (x_i, y_j, z_k) for the m^{th} iteration. The improved value $u^{m+1}_{i,j,k}$ is obtained by solving the discrete interior equation (2.1) with respect to $u_{i,j,k}$, substituting the approximations $u^m_{*,*,*}$ of associated surrounding grid points.

The Gauss-Seidel iteration differs from the Jacobi iteration with respect to a direct substitution: The approximation $u^m_{i,j,k}$ is replaced by a $u^{m+1}_{i,j,k}$ as soon as it is available. Applying a lexicographic ordering of the grid points (this ordering is given by $v^m_l = v^m_{i+j(N_x+1)+k(N_x+1)(N_y+1)} := u^m_{i,j,k}, \quad l = 0..N, \ N = (N_x+1)(N_y+1)(N_z+1) - 1$), we can define the basic step in the nonlinear Gauss-Seidel process as to solve the l^{th} equation

$$g_l[v^{m+1}_1, .., v^{m+1}_{l-1}, v_l, v^m_{l+1}, .., v^m_N] = 0, \tag{2.4}$$

for v_l and to set $v^{m+1}_l = v_l$, where g_l represents the discrete interior equation (2.1) in the l^{th} grid point. Thus, we solve successively equation (2.4) for $l = 0, .., N$ to obtain the $(m+1)^{th}$ approximation.

Gauss-Seidel (Jacobi) iterations converge for linear elliptic problems. A convergence proof is in this case more difficult to establish due to the complicated nonlinearity: It could involve an analysis of the global mapping $G = (g_1, .., g_N)$ in terms of M-functions, see e.g. Ortega en Rheinbold [4]. Results on this are not yet available.

Multiplying equation (2.1) by $-h^2_x h^2_y h^2_z/8$ yields the following cubic equation which forms the equation (2.4) for our case.

$$z^3 + a_2 z^2 + a_1 z + a_0 = 0, \qquad z = u_{i,j,k}. \tag{2.5}$$

The coefficients a_0, a_1 and a_2 are set by the used finite differences, expanding the determinant definition. The roots of the cubic equation are given by the formulas of Cardano

$$\begin{aligned} z_1 &= -a_2/3 + (s_1 + s_2), \\ z_2 &= -a_2/3 - (s_1 + s_2)/2 \quad +\mathbf{i}\sqrt{3}(s_1 - s_2)/2, \quad \mathbf{i} = \sqrt{-1} \\ z_3 &= -a_2/3 - (s_1 + s_2)/2 \quad -\mathbf{i}\sqrt{3}(s_1 - s_2)/2, \end{aligned}$$

with

$$\begin{aligned} s_1 &= \left(r + \sqrt{d}\right)^{1/3}, \qquad & d &= q^3 + r^2, \\ s_2 &= \left(r - \sqrt{d}\right)^{1/3}, \qquad & q &= a_1/3 - a_2^2/9, \quad r = (a_1 a_2 - 3a_0)/6 - a_2^3/27. \end{aligned}$$

The value of d forms a discriminant for the roots: If $d < 0$ then all the roots are real, if $d > 0$ then one of the roots is real and the other two are complex conjugate and, if $d = 0$ then all the roots are real and at least two are equal.

2.3 Convex approximations

In the $d < 0$ case (see previous section), three possibilities appear for the Gauss-Seidel step in a grid point. We will show that this case can occur frequently and that only choosing the smallest root is appropriate for obtaining a convex approximation with the given Gauss-Seidel scheme. It forms a natural extendsion from the one root case.

Locally, a convex approximation must satisfies

$$\begin{aligned} u_{i,j,k} &\leq (u_{i+1,j,k} + u_{i-1,j,k})/2, \\ u_{i,j,k} &\leq (u_{i,j+1,k} + u_{i,j-1,k})/2, \\ u_{i,j,k} &\leq (u_{i,j,k+1} + u_{i,j,k-1})/2. \end{aligned}$$

and therefore

$$u_{i,j,k} \leq [u_{i+1,j,k} + .. + u_{i,j,k-1}]/6 = -a_2/3, \tag{2.6}$$

where a_2 is the coefficient of the cubic discrete interior equation (2.5). Using the determinant definition of the Monge-Ampère operator (1.2) this cubic equation can be represented as

$$-\det(Q - (2z)I_3) + h^6 f_{i,j,k} = 0,$$

with $Q = Q^T$. First, let us we replace the cubic equation by the following characteristic equation

$$-\det(Q - (4w)I_3) = w^3 + a_2 w^2 + a_1 w + (a_0 - h^6 f_{i,j,k}) = 0.$$

This case corresponds to $f_{i,j,k} = 0.0$. The roots of this equation, numbered as $w_1 \leq w_2 \leq w_3$, are real because Q is a symmetric real matrix! According to the formulas of Cardano these roots can be written as

$$w_1 = -a_2/3 + r_1, \quad w_2 = -a_2/3 + r_2, \quad w_3 = -a_2/3 + r_3.$$

A Convex Gauss-Seidel step using a root w_l, $l = 1, 2, 3$ should satisfy the convexity condition (2.6)! Suppose now that all $r_l \geq 0$, $l = 1, 2, 3$, then the following implications hold.

$$w_1 + w_2 + w_3 = -a_2 \;\Rightarrow\; r_1 + r_2 + r_3 = 0 \;\Rightarrow\; r_1 = r_2 = r_3 = 0$$

In this case, local convexity condition (2.6) is either satisfied as an equality, or at least r_1 (w_1 is the smallest root) is negative. Consequently, w_1 always fulfills the given condition.

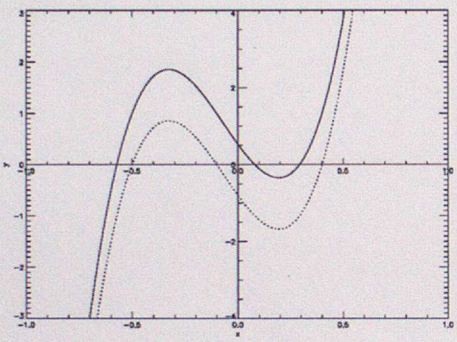

Figure 2.1: Graph of the cubic discrete interior equation (solid line).

Let us return to the original cubic equation (2.5). The graph of this equation is a positive vertical translation over $h^6 f_{i,j,k}$ of the characteristic equation, see Figure (2.1). The smallest cubic root, ($z_1 \leq z_2 \leq z_3$), is smaller than w_1 and consequently it satisfies the local convexity condition . Examples, in which $z_1 < -a_2/3$, $z_2 > -a_2/3$ and $z_3 > -a_2/3$, are easily created and actually appeared frequently in some of the test cases. The graph shows also that the smallest root is the natural extension of the one real case !

Remark: It should be stressed here that the equation (2.6) is only used to make the most sensible choice of the root. It is a nescessary condition because test cases with other root choices showed divergence of the method.

Remark: If the formulas of Cardano are replaced by a Newton method then the considered scheme is called a Newton-SOR method.

2.4 Boundary conditions

The 3D block area has 6 boundary faces. It is possible to have a mixture of different types of boundary conditions, provided that the boundary value problem of the

Monge-Ampère equation is well-posed. The boundary conditions considered are of the Dirichlet, Neumann and mixed type.

Dirichlet boundary conditions are relatively easy to implement. On the boundary of the grid, this is

$$\{(x_i, y_j, z_k) | i = 0, N_z \vee j = 0, N_y \vee k = 0, N_z\},$$

the values of the approximation are prescribed by $u_{i,j,k} = g(x_i, y_j, z_k)$ where g is a specified function. The Gauss-Seidel iterations are executed for $i = 1, .., N_x - 1$, $j = 1, .., N_y - 1$ and $k = 1, .., N_z - 1$.

The Neumann boundary conditions are slightly more complicated. Fictituous boundary points outside the domain are introduced in order to approximate the derivatives on the boundaries by second order central differences. The x,y and z derivatives are specified instead of the normal derivative, because the volume is a 3D block. For example, the normal derivative on the boundary $i = N_x$ is equal to the x-derivative and the value in the fictituous points (x_{N_x+1}, y_j, z_k), $j, k = 0, n$ are prescribed by

$$u_{N_x+1,j,k} = u_{N_x-1,j,k} + 2h_x g_x(x_{N_x}, y_j, z_k).$$

Similarly

$$u_{-1,j,k} = u_{1,j,k} - 2h_x g_x(x_0, y_j, z_k),$$

for the value of u in the fictituous points (x_{-1}, y_j, z_k).

Neumann boundary conditions are degenerate on the ribs of the 3D block because a normal direction is not determined. However, the value of the associated fictituous points can be specified by truncated Taylor series. For example, the implemented second order discretisation related to the rib (x_{N_x}, y_{N_y}, z_k), $k = 0, .., N_z$ and the corner $(x_{N_x}, y_{N_y}, z_{N_z})$ are, respectively,

$$u_{N_x+1,N_y+1,k} = u_{N_x-1,N_y-1,k} + 2h[g_x(x_{N_x}, y_{N_y}, z_k) + g_y(x_{N_x}, y_{N_y}, z_k)]$$

and

$$\begin{aligned} u_{N_x+1,N_y+1,N_z+1} &= u_{N_x-1,N_y-1,N_z-1} + 2h[g_x(x_{N_x}, y_{N_y}, z_{N_z}) + \\ &\quad g_y(x_{N_x}, y_{N_y}, z_{N_z}) + g_z(x_{N_x}, y_{N_y}, z_{N_z})] \end{aligned}$$

The Gauss-Seidel iterations are executed for $i = 0, .., N_x$, $j = 0, .., N_y$ and $k = 0, .., N_z$.

The mixed boundary conditions can be implemented using the same techniques. The specified function g depends also on the previous approximation $u_{i,j,k}$ in the case fictituous boundary points are used to specify boundary derivatives (see Neumann type). Alternatively, g depends also on an approximated value of the normal derivative in

the case values on the boundary itself are specified (see Dirichlet type). The first type of mixed boundaries have been implemented.

Remark: The use of second order differences to specify fictituous boundary points makes the discrete representation of the Monge-Ampère operator drop to first order due to the product of the differencesin the definition! The fictitous points might have to be set by multi-step differences to overcome this problem.

Remark: The use of solely Neumann boundary conditions allows the solution to vary with respect to a constant. As a consequence the numerical scheme might not convergence, unless we add a loop to fix the solution to a specified level.

2.5 Overrelaxation and sweeping

Overrelaxation improves the convergence rate of the relaxation method. The new value in a Gauss-Seidel step is weighted with the old value rather than just replaced by the calculated smallest root z, see equation (2.5).

$$u_{i,j,k}^{m+1} = u_{i,j,k}^{m} + \omega(z - u_{i,j,k}^{m}), \qquad \omega \geq 1$$

In the test cases, overrelaxation is applied with $1.0 \leq \omega \leq 2.0$ for the Dirichlet boundary conditions. Too high values of ω cause divergence. The optimal overrelaxations are found by trial and error (≈ 1.7). A more advanced method for finding the optimal overrelaxation is hard to establish because the discrete system is nonlinear. The convergence rate improved in some test cases upto a factor 4 due to overrelaxation only, see figure (2.5).

The naive way to sweep through the three dimensional grid is a lexicographic update of the three indices i, j and k. However, convergence might improve substantially by an alternative sweeping method. The implemented alternative initially spirals through boundary face $(1, j, k)$, $j = 1, ..N_y - 1, k = 1, .., N_z - 1$ (Dirichlet boundary conditions), starting in the corner $(1, 1, 1)$ and ending in the middle of the surface. The other faces of the boundary are swept similarly. After the boundary is done the next layers are swept in the same manner until the center of the cubic grid is reached.

2.6 Multigrid technique

The use of nested grids also reduces the number of iterations needed to reach a converged stage. For a general introduction to multigrid techniques we refer here to the first chapters of Stüben and Trottenberg [7]. We have to implement a Full Approximation scheme (FAS), because the discrete equations are nonlinear. A V-cycle of the FAS is given by the following steps, where the subscripts h and H represent respectively the fine and the subsequent coarse grid.

1. Make relaxations on the fine grid. $(M_h[u_h] = f_h)$

2. Compute the defect. $(d_h = f_h - M_h[u_h])$
3. Restrict the defect. $(d_h \rightarrow d_H)$
4. Restrict the approximation. $(u_h \rightarrow u_H)$
5. Approximate the right-hand-side. $(f_H \approx d_H + M_H[u_H])$
6. Compute an coarse approximation w_H of $M_H[w_H] = f_H$. In this step we recursively start this FAS scheme !
7. Compute the correction. $(v_H = w_H - u_H)$
8. Interpolate the correction. $(v_H \rightarrow v_h)$
9. Compute the corrected approximation. $(u_h = v_h + u_h)$
10. Make again relaxations on the fine grid. $(M_h[u_h] = f_h)$

Full Multigridding (FMG) means another loop on top of the FAS. It is defined to start at the coarsest grid applying a FAS and the obtained approximation is interpolated to a finer grid to apply again a FAS scheme, until the finest grid is reached.

The most successful way of resticting appeared to be a straight injection, but weighted restrictions do work as well. The FAS scheme converges only if the starting approximation is near the solution because of the validity of the defect approximation of the right-hand-side in step 5. This step can be replaced by a direct restriction of the right-hand-side. This adjusted FAS scheme converges quite well away from the solution which is useful to obtain good initial FAS approximations. In general, initial approximations obtained by the FMG scheme (interpolated converged FAS solutions of a coarser grid) are in general sufficient! The FMG starts really to pay off for the fine grids ($64 \times 64 \times 64$). Other multigrid techniques (for example W-cycles) are not considered.

2.7 Test function results

The developed Monge-Ampère solver is applied to exact known boundary value problems to analyse the numerical behaviour. These problems should fulfill certain requirements to form genuine convex Monge-Ampère problems. First, the right hand side of the Monge-Ampère equation should be positive to form an elliptic problem. Second, the test problems should be based on a convex solution otherwise the choice of the smallest root is not applicable. Third, the cross derivatives terms should be more or less of the same order of magnitude as the homogeneous second derivatives. The following test functions fulfill these requirements for reasonable values of the constants. The plots show the second order convergence of the method for test function (2.7) in the maximum norm and the influence of overrelaxation and alternative

sweeping. Table(2.8) compared to table (??) shows the improved performance by an interpolated initial approximation and the FMG multigrid technique.

$$\begin{aligned} u(x,y,z) &= c_1[c_2x^2 + c_3y^2 + c_4z^2] \\ &\quad +c_5[c_6x^4 + c_7y^4 + c_8z^4] \\ &\quad +c_9\exp[c_{10}x^2 + c_{11}y^2 + c_{12}z^2 + c_{13}x + c_{14}y + c_{15}z] \\ &\quad +c_{16}\sin[c_{17}x^2 + c_{18}y^2 + c_{19}z^2 + c_{20}x + c_{21}y + c_{22}z] \\ &\quad +c_{23}\cos[c_{24}x^2 + c_{25}y^2 + c_{26}z^2 + c_{27}x + c_{28}y + c_{29}z] \end{aligned} \tag{2.7}$$

$$\begin{aligned} u(x,y,z) &= c_1[c_2x^2 + c_3y^2 + c_4z^2] \\ &\quad +c_5\log[c_6 + c_7x + c_8y + c_9z] \\ &\quad \exp[c_{10}x^2 + c_{11}y^2 + c_{12}z^2 + c_{13}x + c_{14}y + c_{15}z] \end{aligned} \tag{2.8}$$

$$\begin{aligned} u(x,y,z) &= [x^2 + y^2 + z^2]/2 \\ &\quad +2\log[10 - x - y - 2z]\exp[(3x^2 + y^2 + z^2 + 2x + y + z)/10] \\ &\quad -2\log[10 - 3x - 2y - z] \\ &\quad +\exp[(8x^2 + 2y^2 + 8z^2 + 4x + 4y + z)/40]/2 \end{aligned} \tag{2.9}$$

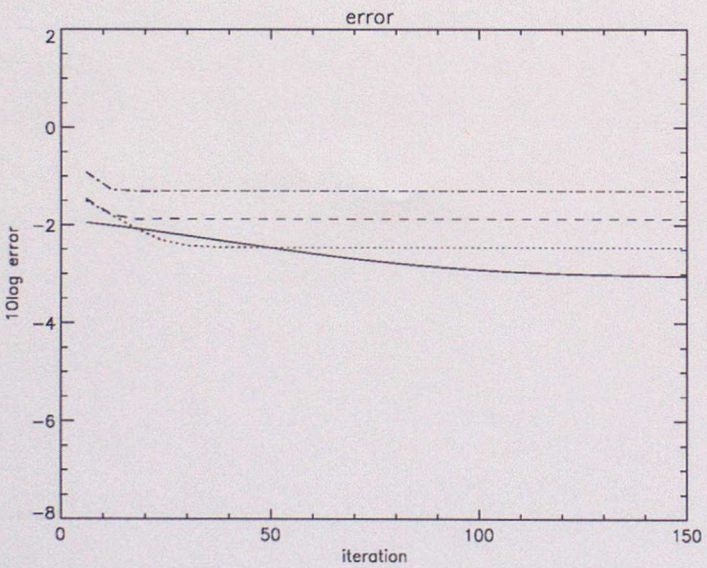

Figure 2.2: **CONVERGENCE DIRICHLET.** Convergence of the solution in the $||\ ||_\infty$ norm. Dirichlet boundary conditions only. The $n \times n \times n$ grids, $n = 32, 16, 8, 4$ are indicated by respectively the solid, dotted, dashed and dash dot line.

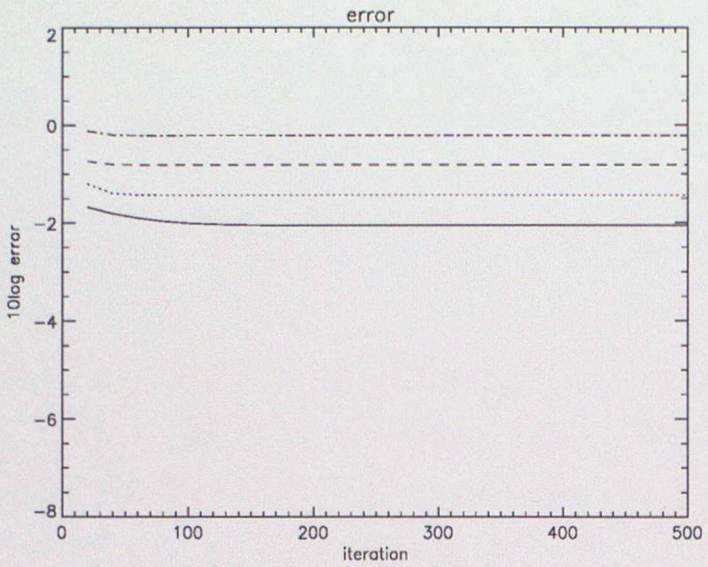

Figure 2.3: **CONVERGENCE NEUMANN.** Convergence of the solution in the $||\ ||_\infty$ norm. Neumann boundary conditions only. The $n \times n \times n$ grids, $n = 32, 16, 8, 4$ are indicated by respectively the solid, dotted, dashed and dash dot line.

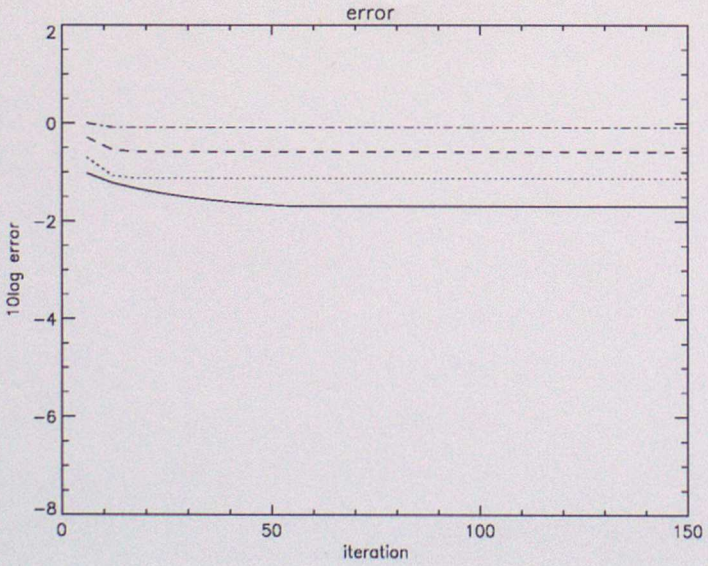

Figure 2.4: **CONVERGENCE GRADIENT.** Convergence of the gradient of the solution in the$||\ ||_\infty$ norm. Dirichlet boundary conditions only. The $n \times n \times n$ grids, $n = 32, 16, 8, 4$ are indicated by respectively the solid, dotted, dashed and dash dot line.

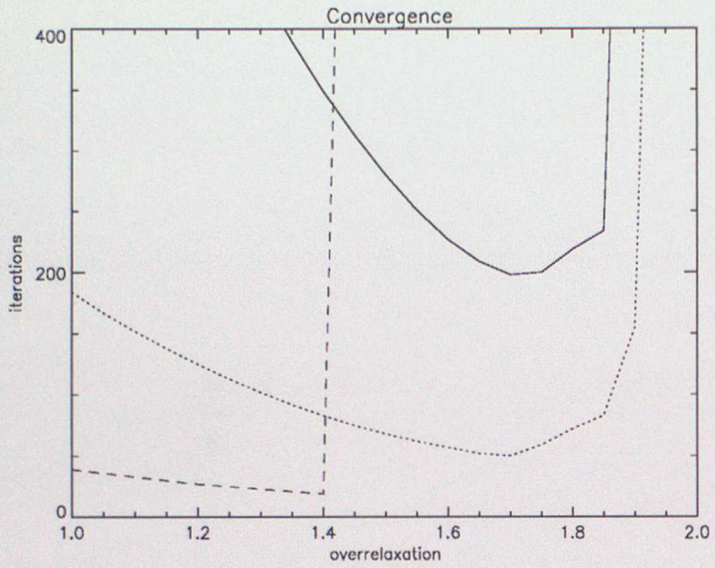

Figure 2.5: **OVERRELAXATION.** Number of iterations necessary to converge to a 0.01 error ($||\ \ ||_\infty$ norm.) in the approximation versus the overrelaxation factor. The $n \times n \times n$ grids, $n = 32, 16, 8$ are respectively indicated by the solid,dotted and dashed line.

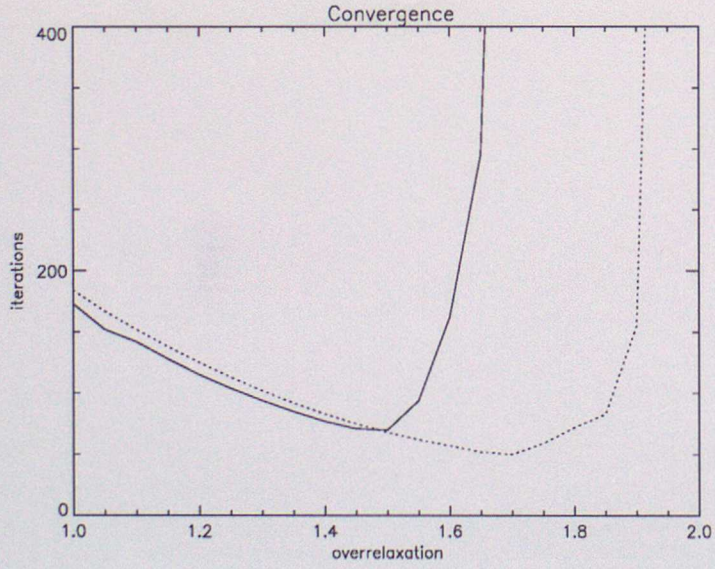

Figure 2.6: **SWEEPING.** Number of iterations necessary to converge to a 0.01 error ($||\ \ ||_\infty$ norm.) in the approximation versus the overrelaxation factor. $16 \times 16 \times 16$ grid. The alternative and lexicographice sweeping methods are respectively indicated by the solid and dotted line.

SINGLE OVERRELAXATIONS testfunction (2.9)

$.3664 \leq u \leq 8.567$,		$5.798 \leq f \leq 44.32$,		$.2255 \leq u_z \leq 4.690$
$-3.227 \leq u_x \leq 5.948$,		$-1.681 \leq u_y \leq 2.465$,		
Grid: N^3	$\|\|u-v\|\|_\infty$ $\times 10^{-3}$	$\|(u-v)(\mathbf{x}_c)\|$ $\times 10^{-3}$	$\|\|M_h[u]-f\|\|_\infty$ $\times 10^{-4}$	Iterations:	Cubic swaps: $\times 10^6$
8	3.7284	3.5897	0.0021	48	0.016
16	0.9554	0.9172	0.0238	174	0.587
32	0.2339	0.2229	19.685	540	16.08
64	0.0641	0.0611	1451.8	2300	526.6

Figure 2.7:

SOR vs. MULTI-GRID testfunction (2.9)

		single relaxations:			Full multigrid scheme:	
Grid: N^3	$\|(u-v)(\mathbf{x}_c)\|$ 10^{-3}	Iterations:	Cubic swaps: $\times 10^6$	$\|(u-v)(\mathbf{x}_c)\|$ 10^{-3}	V-cycles:	Cubic swaps: $\times 10^6$
8	3.5897	48	0.016	3.5897	8	0.018
16	0.9172	132	0.466	0.9172	7	0.184
32	0.2333	264	8.371	0.2305	6	1.458
64	0.0506	444	122.6	0.0575	6	12.47

Figure 2.8:

Chapter 3

PV inversion in semi-geostrophic theory

Potential vorticity (PV) is an important diagnostic for atmospheric flow. Spatial distributions of PV are derived from both operational weather analyses and idealized theoretical models. The theory of PV shows how the distribution and movement of PV control, through (mutual) adjustment, dynamical evolutions of the atmosphere. See for an exhaustive overview the article of Hoskins, McIntyre and Robertson [3]. A semi-geostrophic balance can be used to analyse the advection of PV. Consider the following PV conserving semi-geostrophic balance developed by Cullen et al. [1].

$$\begin{cases} \dfrac{Du_g}{Dt} - f(v - v_g) &= 0, \\ \dfrac{Dv_g}{Dt} + f(u - u_g) &= 0, \end{cases} \tag{3.1}$$

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0, \tag{3.2}$$

$$\frac{D\theta}{Dt} = 0, \tag{3.3}$$

$$\nabla_x \phi = \left(f v_g, -f u_g, g\frac{\theta}{\theta_0} \right). \tag{3.4}$$

A relatively straightforward result (see Roulstone and Norbury [6]) which involves the Hamiltonian form of the equations of motion, shows that the Jacobian of a geostrophic coordinate transformation, this is

$$\mathbf{X} = (X, Y, Z) = (x + v_g/f, y - v_g/f, \frac{g\theta}{f^2\theta_0}), \tag{3.5}$$

is equal to the potential vorticity ,

$$q = \frac{g}{f\theta_0}\left[\frac{\zeta \cdot \nabla_x \theta}{\rho}\right] = \frac{\partial \mathbf{X}}{\partial \mathbf{x}}. \tag{3.6}$$

If a potential P is introduced such that $\mathbf{X} = \nabla_x P$ then a nonlinear elliptic Monge-Ampère equation is obtained

$$M[P] := \det(H_x(P)) =$$

$$P_{xx}P_{yy}P_{zz} - P_{xx}P_{yz}^2 - P_{yy}P_{xz}^2 - P_{zz}P_{xy}^2 + 2P_{xy}P_{xz}P_{yz} = q, \tag{3.7}$$

where $H_x(P)$ is the Hessian matrix of P. This equation can be called the invertibility relation for the semi-geostrophic balance, because solutions allow us to recover wind and temperature fields from a spatial distribution of q

$$u_g = f(\frac{\partial P}{\partial x} - x), \quad v_g = f(y - \frac{\partial P}{\partial y}), \quad \theta = \frac{f^2\theta_0}{g}\frac{\partial P}{\partial z}. \tag{3.8}$$

In physical space ($\mathbf{x}$), semi-geostrophic PV inversion using equation (3.7) give rise to serious complications with respect to the specification of meaningful boundary conditions. The overcome these troubles, a potential R can be introduced for the physical coordinates,

$$\mathbf{x} = (x, y, z) = \nabla_X R$$

In this way, the invertibility relation can be transformed into a Monge-Ampère equation in (the so-called) dual space.

$$\det(H_X(R)) = \rho, \qquad \rho(X, Y, Z) = q^{-1}(x, y, z),$$

where $H_X(R)$ is the Hessian matrix of R with respect to $\mathbf{X}$. ρ is a pseudo-density of which the volume integral over the domain in dual space must be equal to the volume of the considered part of the atmosphere e.g.

$$\nabla_X(R) \in \partial\Omega, \qquad |\mathbf{X}| \to \infty$$

This problem is known as the second boundary value problem, (see e.g. Pogorelov [5] for 2D cases) and appears naturally from the energy minimisation principle on which the given semi-geostrophic balance is based. This principle also implies that only convex solutions of the Monge-Ampère equation are of interest with respect to semi-geostrophic theory.

In short, it is essential for numerical PV or pseudo-density inversion to develop a robust and fast Monge-Ampère equation solver. A finite difference approach can become very successful when meaningful boundary conditions can be included properly. This report elaborates the preliminary stage of Dirichlet value problems and contains some experiments with Neumann and mixed boundary conditions. Future research might concentrate on the possibilities to include the Dirichlet bounday value problems in an boundary adjusting loop solving the appropriate set of boundary conditions.

3.1 PV anomalies

A PV anomaly is a part of the volume with a different PV value from its surroundings. This may be an part of relatively high PV in a region of relatively low PV or vice versa. Thorpe [8], [9] describes models for typical circular symmetric cut-off PV anomalies in the troposphere and near the tropopause. We simplify these models to apply the Monge-Ampère solutions for the recovery of the wind and temperature fields. The first considered anomaly is characterized by an isolated circular symmetric patch of high PV in the troposphere, the second one by a stratospheric circular symmetric tongue of high PV which indents into the troposphere. A third anomaly is an equivalent indentation of the tropospheric low PV into the stratosphere. In these cases, the influence of the boundaries are restricted, because the anomalies have only a bounded effect in the atmosphere. In other words, away from the anomaly the horizontal wind field $(u_g, v_g, 0)$ tends to zero and the potential temperature tends to decrease linearly with height. The problems are made non-dimensional by the typical synoptic scales

$$
\begin{array}{lcll}
x & = & Lx', & L = 1000 \text{ km}, \\
y & = & Ly', & \\
z & = & Hz', & H = 10 \text{ km}, \\
\mathbf{v} & = & NH\mathbf{v}', & N = 10^{-2} 1/\text{s}, \\
q & = & N^2/\rho q', & \rho \approx 1 \text{kg/m}^3 \\
\theta & = & N^2 H/g\theta_0\theta', & \theta_0 = 300\text{K}, \quad \text{N}^2/\text{g} = 0.1.
\end{array}
$$

The accents of the non-dimensional variables are again skipped for convenience.

3.2 Axi-symmetric cases.

For the axi-symmetric tropospheric anomaly, the spatial distribution of PV is specified by

$$
\begin{cases}
q = 1 + q_0 \dfrac{\cos(\pi(z-1))}{(1+R^2/R_0^2)^2}, & z \in [\frac{1}{2}, 1\frac{1}{2}], \\
q = 1, & \text{otherwise},
\end{cases}
\tag{3.9}
$$

where $R^2 = x^2 + y^2$ and $R_0 \approx 0.4$. The atmospheric volume is defined by the cube $\Omega = [-1,1] \times [-1,1] \times [0,2]$. $q_0 = 9$ corresponds to a realistic value of $q = 10$ in the center of the patch, and $q \approx 1$ outside the blob. We use again central differences to make the actual recovery of the wind and temperature fields which in these non-dimensional case is given by

$$u_g = \frac{\partial u}{\partial x} - x, \quad v_g = y - \frac{\partial u}{\partial y}, \quad \theta = \frac{\partial u}{\partial z}.$$

The required zero horizontal wind field $(u_g, v_g, 0)$ and the constant lapse rate are set by, respectively, the following Dirichlet or Neumann boundary conditions

$$\begin{aligned} u(x,y,z) &= (x^2 + y^2 + z^2)/2, \\ u_x(x,y,z) &= x, \qquad\qquad \mathbf{x} \in \partial\Omega, \\ u_y(x,y,z) &= y, \\ u_z(x,y,z) &= z. \end{aligned}$$

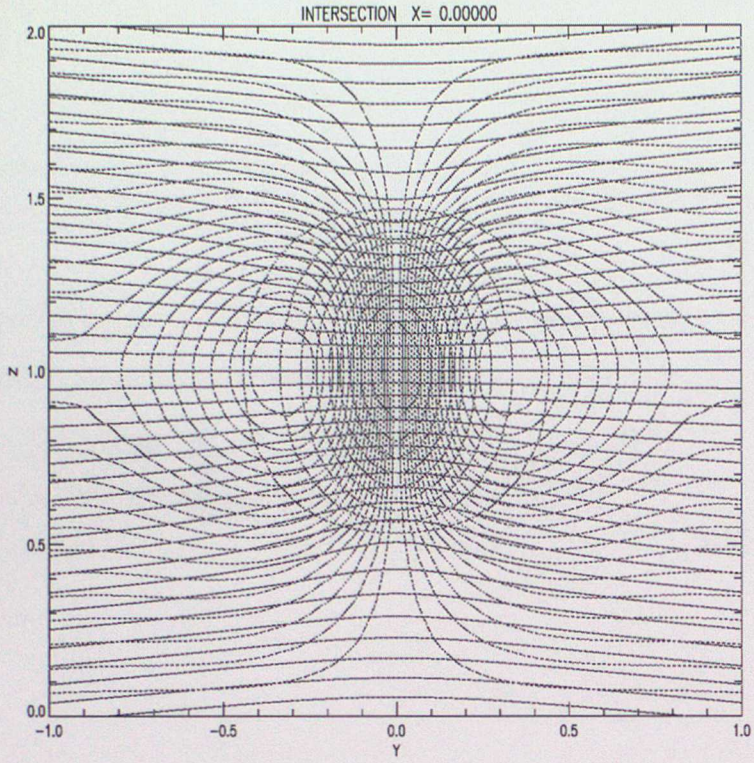

Figure 3.1: Axi-symmetric troposphere anomaly. Vertical intersection of the vortex structure at $x = 0$. $32 \times 32 \times 32$ grid. Dirichlet boundary conditions.

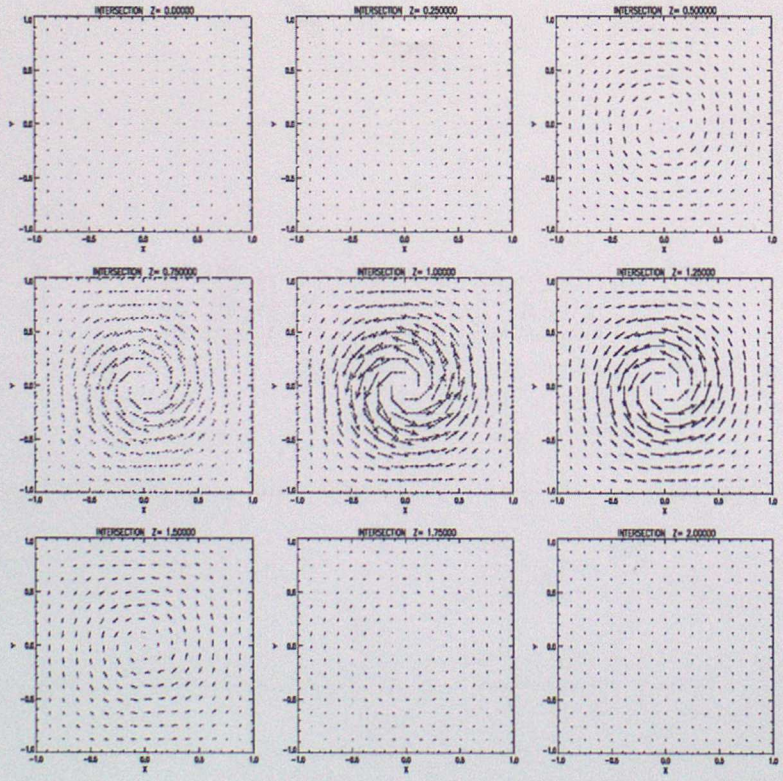

Figure 3.2: Idem, for horizontal intersections

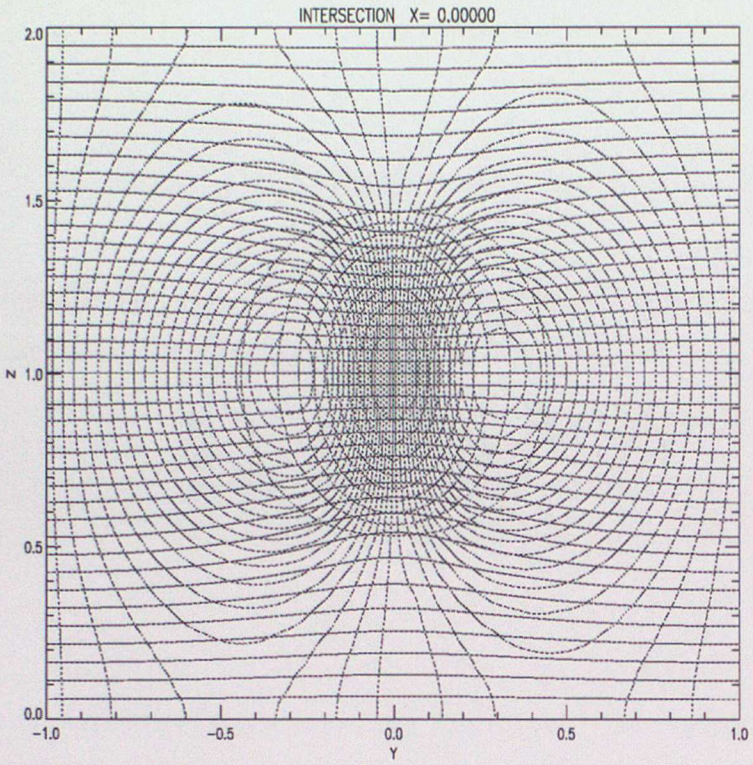

Figure 3.3: Idem, Neumann Boundary conditions.

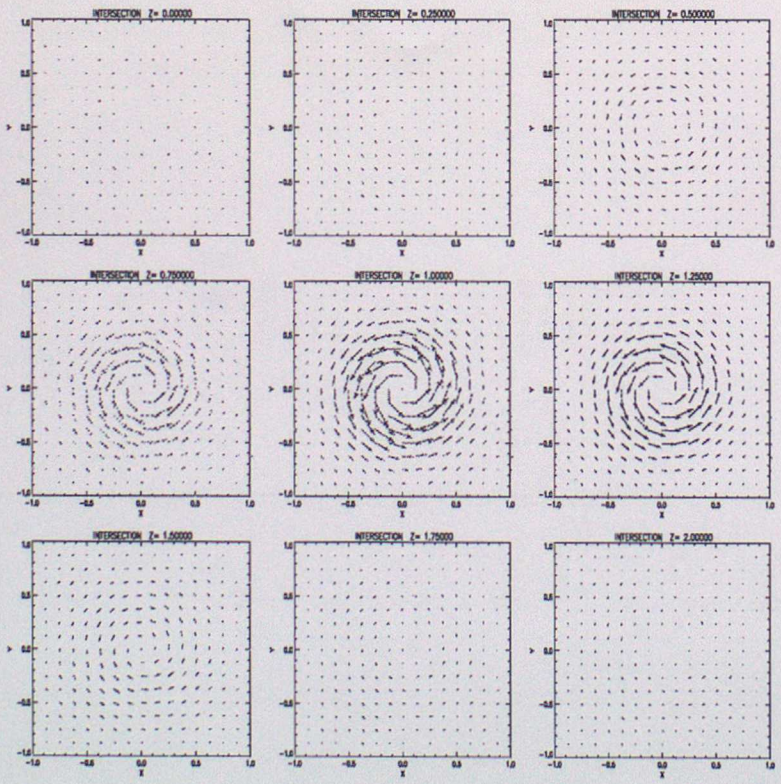

Figure 3.4: Idem, Neumann Boundary conditions.

We restrict ourselves to $32 \times 32 \times 32$ grids and Dirichlet boundary conditions. For the tropopause anomaly, which indents into the troposphere, the spatial distribution of PV is specified by

$$\begin{cases} q = q_2, & z \geq Z_0 - Z_c \cos(\pi/2R/R_0), & R \leq R_0, \\ q = q_2, & z \geq Z_0, & R > R_0, \\ q = q_1, & \text{otherwise}, \end{cases} \tag{3.10}$$

where $R^2 = x^2 + y^2$ and $R_0 \approx 0.4$. (This is a simplified version of Thorpe's [9] distribution.) The atmosphere is defined by the cube $\Omega = [-1,1] \times [-1,1] \times [0,2]$. q_2 is the high PV value of the stratosphere, and q_1 is low PV value of the troposphere. Z_0 is the unperturbed position of the tropopause. On the boundaries, we require again an zero horizontal wind field $(u_g, v_g, 0)$ and a constant lapse rate in both the troposphere and the stratosphere. For the lower stratosphere a linear term is added is to adjust the lapse rates. The Dirichlet or Neumann boundary conditions are set by, respectively,

$$\begin{aligned} u(x,y,z) &= (x^2+y^2+cz^2)/2 \\ &\quad +(1-c)(Z_0 z + Z_0^2), && z \geq Z_0 \\ u(x,y,z) &= (x^2+y^2+z^2)/2, && z < Z_0 \\ \\ u_x(x,y,z) &= x, && \mathbf{x} \in \partial\Omega, \\ u_y(x,y,z) &= y, \\ u_z(x,y,z) &= cz, && z \geq Z_0, \\ u_z(x,y,z) &= z, && z < Z_0, \end{aligned} \tag{3.11}$$

where Z_0 represents the tropopause height.

An anticyclonic motion is obtained if a tongue of tropospheric low PV indents in the high PV stratosphere. The description of this PV anomaly is identical to the previous one except for the value of Z_c in equation (3.10) which has to be negative.

3.3 A non-symmetric case.

To show the essential nonlinearity of the solution method a more dramatic case is added. We defined a tropopause anomaly model similar to the previous one but with a nonsymmetric irregular shaped tropopause. the results for a $32 \times 64 \times 32$ grid and Dirichlet boundary conditions are presented in figure 3.9 and 3.10

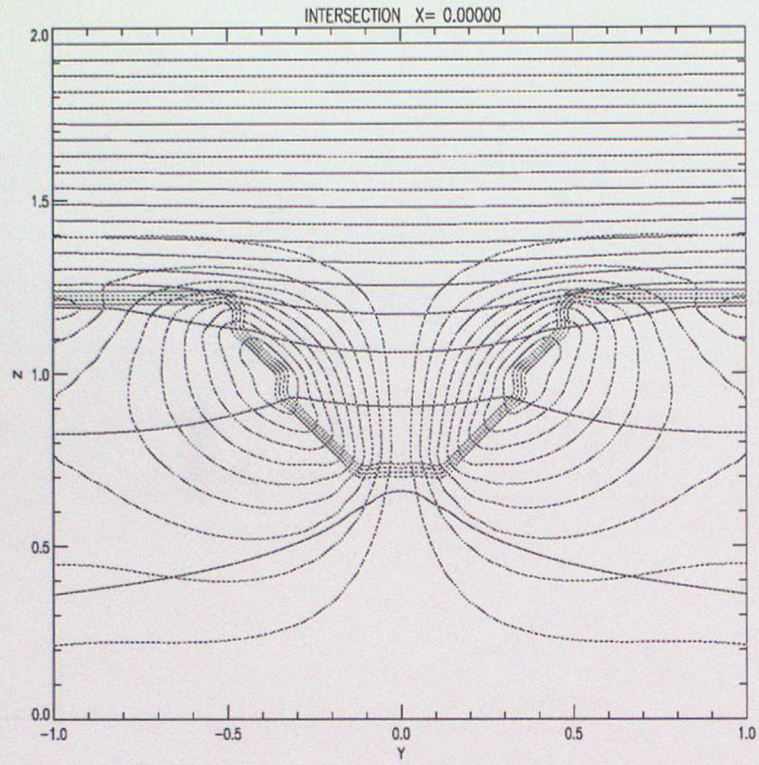

Figure 3.5: Vertical intersection of a warm core upper cyclone

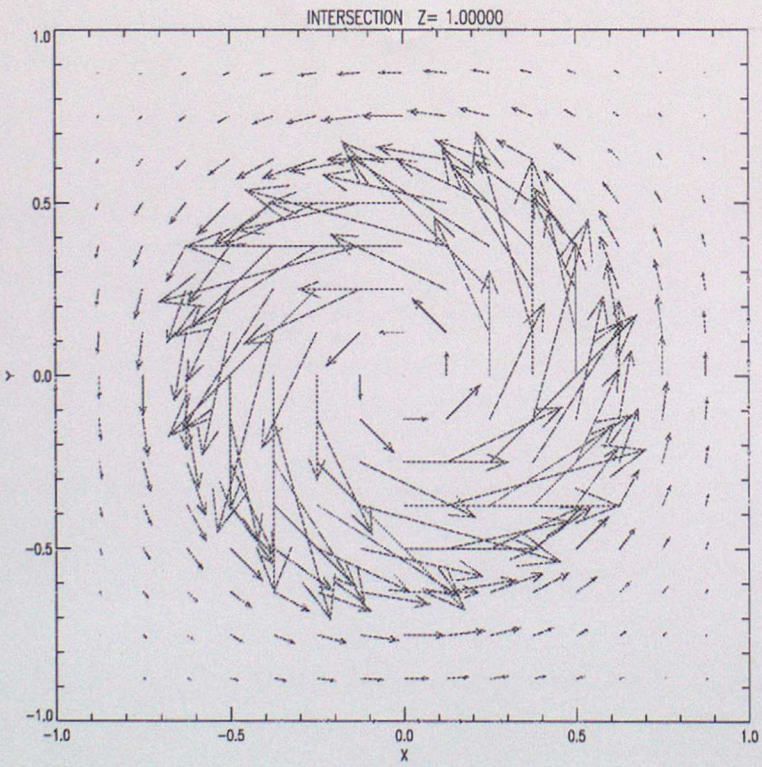

Figure 3.6: Horizontal intersection of a warm core upper cyclone

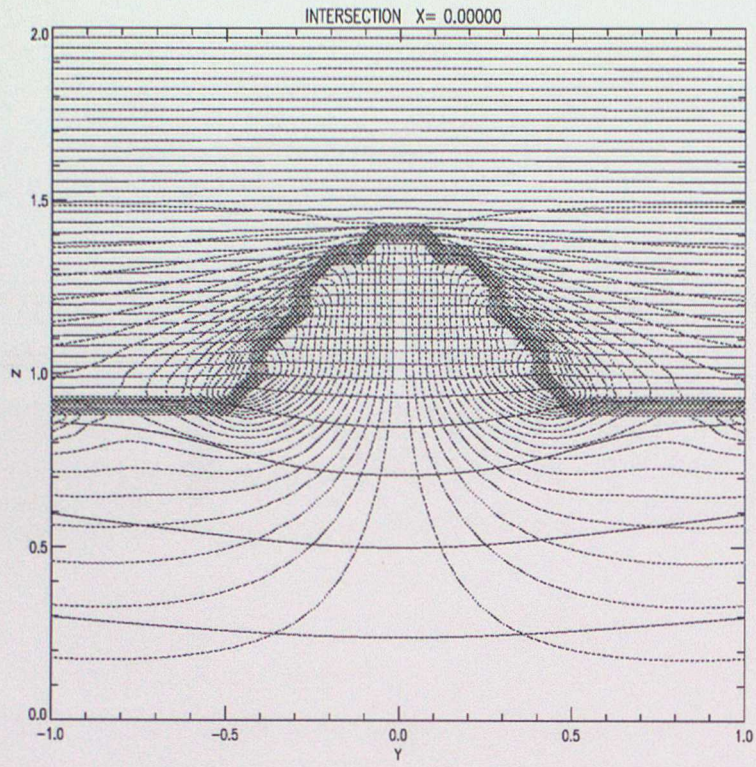

Figure 3.7: Vertical intersection of a cold core upper anti-cyclone

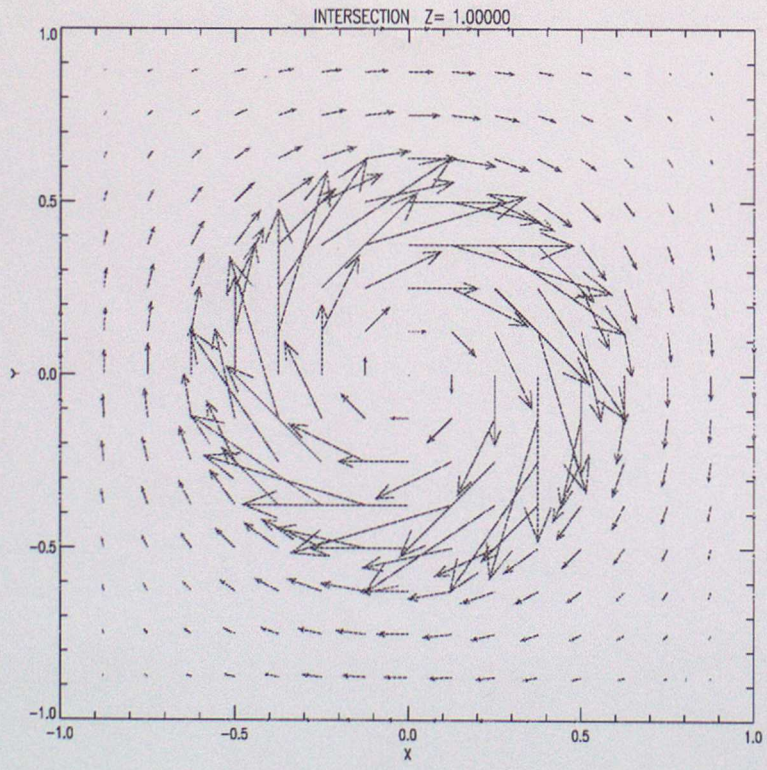

Figure 3.8: Horizontal intersection of a cold core upper anti-cyclone

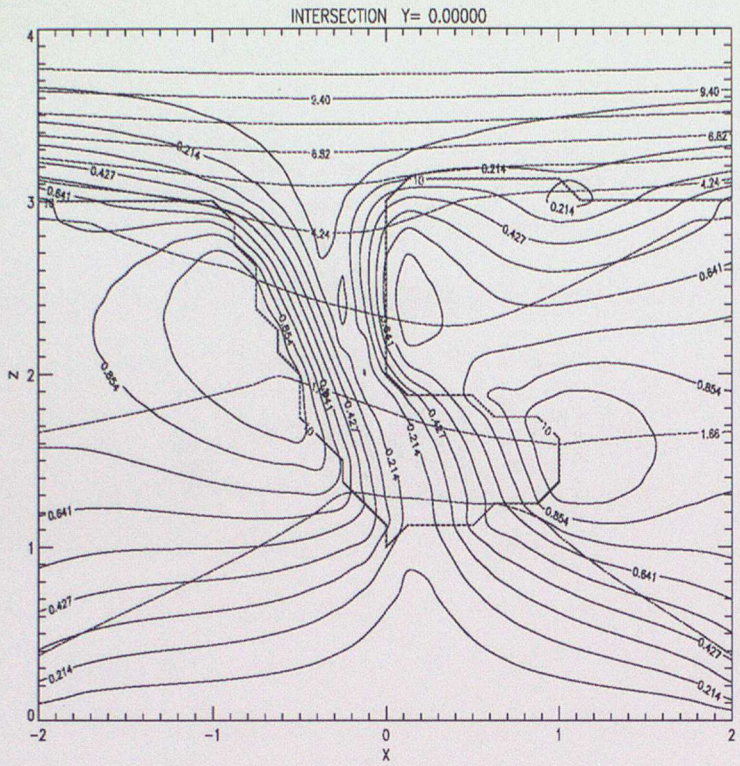

Figure 3.9: Vertical intersection of a warm core upper cyclone

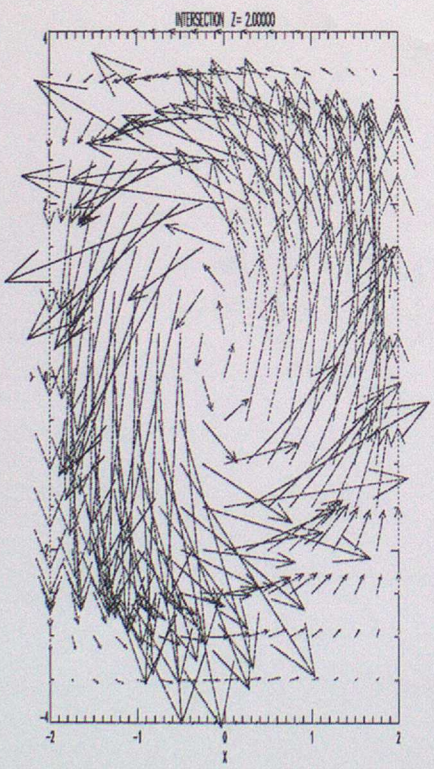

Figure 3.10: Horizontal intersection of a warm core upper cyclone

Chapter 4

The FORTRAN 77 implementation

The numerical method is implemented in FORTRAN 77 routines which will be described in this chapter. The output of the code can be handled by the PV-wave graphical package. We made several PV wave procedures to draw pictures on the HP terminals and save pictures in postscript format.

4.1 Program outline

The considered FORTRAN 77 program is stored in the following files. Of each file a small description is given of the purposes of its subroutines.

mas.f Main program. After initializing the boundary value problem, the main program calls the Full Multigrid Grid loop which can be degenerated to a Full Approximation Scheme with repeated V-cycles or even further to a single grid relaxation method .

relax.f The relaxation procedures make nonlinear Gauss-Seidel iterations with the chosen sweeping method and difference molecule by solving cubic equations.

boundary.f Boundary data have to be set initially and For Neumann or mixed typed boundaries data (the fictituous points) have to be adjusted after every relaxation sweep.

grid.f The relaxation and multigrid methods require change of dimensions and the set-up, restriction and interpolation of grids.

misc.f The operation of the numerical method require several miscelleaneous routines such as elementary matrix operations, derivative approximations using finite differences (to specify the gradient) and the printing of errors and solution (output).

prob*.f Every application of the numerical method needs the specification of grid-sizes, right hand sides, boundary conditions initial approxiamtions and additional parameters.

func.f In this file a library of functions is stored which can be used to construct test functions and boundary conditions.

4.2 Data structure

rect Array of reals containing the 3D block volume.

x,y,z Arrays of reals containing grid the coordinates of the grid points.

c Array of reals containing constants used for the definition of the boundary value problems.

sm Work space. Array of reals containing approximations, derivatives, right-hand sides defects, etc.

xc Array of reals containing parameters controlling the performance of the method.

ic Array of integer containing the boundary specification and integer parameters controlling the performance of the method.

nc Array of integer containing grid dimensions, multi grid levels and difference molecule increments.

uc Array of integer containing approximation positions.

vc Array of integer containing exact solution positions.

Like in many FORTRAN 77 programs, the calculations are made in one big 1 dimensional array **sm** of reals (double precision) which forms the work space. The individual approximated values in the grid points are accessed by integer increments on the index of this big !D array. To control this integer calculations with the proper increments, an array of integer type (**nc**) is set, containing the essential information about the dimensions of the 3D grids, fictituous boundary points, sweeping limits, multigrid depths and the finite difference molecules.

nc(1) Grid dimension x-direction. ($= 4 * 2^{nc(81)}$)

nc(2) Grid dimension y-direction. ($= 4 * 2^{nc(82)}$)

nc(3) Grid dimension z-direction. ($= 4 * 2^{nc(83)}$)

nc(4) number of points in x-direction. ($= nc(1) + 1$)

nc(5) number of points in y-direction. ($= nc(2) + 1$)

nc(6) number of points in z-direction. ($= nc(3) + 1$)

etc.

The positions in the array **sm** of the first point of approximated grid solutions are set in the integer array **uc**. For example, the first point of the approximation u is given by (uc(1) + 1). In the same way, the start positions in **sm** of the exact grid solution and its gradient are set in the integer array **vc**.

uc(1) Approximation u on finest grid.

uc(2) Approximated derivative u_x on finest grid.

uc(3) Approximated derivative u_y on finest grid.

uc(4) Approximated derivative u_z on finest grid.

uc(5) Right-hand-side on finest grid.

uc(>5) Contain approximations, defects and right hand sides on coarser grids used in the multi grid scheme.

vc(1) Exact solution U on finest grid.

vc(2) Exact derivative U_x on finest grid.

vc(3) Exact derivative U_y on finest grid.

vc(4) Exact derivative U_z on finest grid.

There are several parameters controlling the performance the program. Parameters of integer type are stored in the array **ic** and parameters of real type are stored in **xc**. It would be too much to explain every individual parameter at this stage. Besides this, many parameter are derived from others to enlarge the efficiency of the method. The most important ones are given here.

ic(1) Number of levels in FMG scheme.

ic(2) Print the solution.

ic(3) Solution method.

ic(5) Sweeping method.

ic(6) Choice finite difference molecule.

ic(7) Existence exact solution.

ic(21) Boundary condition $x = x_0$. 1 = Dirichlet. 2 = Neumann. 3 = mixed.

ic(22) Boundary condition $x = x_1$.

ic(23) Boundary condition $y = y_0$.

ic(24) Boundary condition $y = y_1$.

ic(25) Boundary condition $z = z_0$.

ic(26) Boundary condition $z = z_1$.

xc(1) Grid size x-direction. ($= 1/nc(1)$)

xc(2) Grid size y-direction. ($= 1/nc(2)$)

xc(3) Grid size z-direction. ($= 1/nc(3)$)

xc(20) Overrelaxation factor, ω.

xc(21) $1 - \omega$

4.3 Procedures

In this section a will describe all the subroutines in a few lines.

Routines of mas.f

mas Main program which allocates the memory usage, initalizes the boundary value problem and calls the Full multi grid routine.

FMG Full multigrid scheme which loops over a repeated FAS scheme.

RVC Repetition of a FAS scheme (FAS, FASP) which can be degenerated to a single relaxation method.

FASP Recursive routine which contains the restriction and prolongation of a original Multigrid V-cycle.

FASP Recursive routine which contains the restriction and prolongation of a modified Multigrid V-cycle.

Routines of prob*.f

parameters Set the parameters and dimensions stored in **ic**,**nc** and **xc**.

constants Set the constants **c** used in the several functions to describe the problem. (NB! c(70) is used to pass the solution u in the Robin boundary value cases.)

start Define the start solution .

exact Define the exact solution .

righthandside Define the right-hand-side function.

xminbound Define start boundary condition , Dirichlet Neumann or Robin for the boundary $x = x_0$.

xmaxbound Idem.

yminbound Idem.

ymaxbound Idem.

zminbound Idem.

zmaxbound Idem.

Routines of relax.f

relax Make a relaxation step.

toprelax Make a relaxation step on finest grid and adjust the boundary if required.

sweep1 Lexicographic sweeping.

sweep2 Alternative 'spiralling' sweeping.

update1 Set the cubic equation according to a difference molecule.

update2 Set the cubic equation according to another difference molecule.

cubicroot calculation of the smallest root by means of the formulae of Gordano.

cubicroottest Not really used. To test the cubicroot routine.

Routines of func.f

funcrhs Calculate the right-hand side of a specified exact test function.

det Calculate the determinant of a 3D symmetric matrix from its entries.

prodfunc Calculate the product of 2 functions inluding the associated derivatives, which are set be the product rule.

linxyz Function with logaritmic terms.

sqxyz Function with 2-order polynomial terms.

quadxyz Function with 4-order polynomial terms.

exppol Function with exponential terms.

logpol Function with logaritmic terms.

sinpol Function with sine terms.

cospol Function with cosine terms.

vertbound

Routines of grid.f

setupgrid Set up the grid spacing etc.

goto-fine-grid Adjust the dimensions and increments according to the subsequent finer grid.

coarse2fine Interpolate a approximation to subsequent finer grid.

interpolation Called by coarse2fine

goto-coarse-grid Adjust the dimensions and increments according to the subsequent coarser grid.

fine2coarse Weighted Restriction of an approximation to subsequent coarser grid.

inject Straight injection of an approximation to subsequent coarser grid.

getdefect Calculate a defect.

makerhs Calculate the multi-grid approximation of the right-hand-side.

Routines of misc.f

getmem Allocate part of the workspace for an approximation

fillmat Fill a approximation with a specified function.

copymat Make a copy of an approximation.

substractmat Subtracting 2 approximations.

adjustmat MG: Adjust an approximation.

grad-and-test Combine gradient calculations and testing.

grad Calculate the gradient using finite differences.

compare Compare approximation with exact known solution.

testrhs Calculate the defect between the right-hand-side and a approximation of the right-hand-side using finite differences.

printerror Write the specified errors to the output.

printsol Write the solution to the output field.

openfiles Open output.

closefiles Close output.

hello Banner.

message-problem Output banner.

Routines of boundary.f

keepmat Level the approximation in case of all Neumann boundary conditions.

setup-boundaries Fill in fixed boundary data.

adjustboundary Adjust fictituous boundary points for Neumann and mixed boundary data. (NB! notice the special role of constant c(70).)

4.4 Input/output

Input The program doesn't have an explicit input. To set the parameters, right-hand-sides and boundary conditions the routines in a file prob*.f has to be altered and linked to the rest of the program. In other words running the program starts with editing this file.

Output The output consist of 3 files. The first file **data/ma.out** contains data of the decreasing errors during the iteration. This file can be viewed directly to analye the performance of the method. A part of this output is also sent to the screen during the iteration. Second, the error information is also written to the file **data/error.out** which is used to draw graphs with PV wave. Thrid, to make horizontal and vertical intersections together with wind fields the approximation, its gradient and the associated right-hanb-side and grid are written to a file **data/control.dat**. This file is read by the PV-wave routine **getdata** to make the actual graphical output.

4.5 PV-Wave

The PV-Wave routines for the graphical output associated with this Fortran 77 program are stored in the file **MAgraph.pro** Again we will discuss every procedure briefly.

GETDATA Read the data produced by the FORTRAN 77 code to solve the Monge-Ampère equation. Calculate also the associated gradient wind field!

OPENWIND Open a window on the screen or a .ps file for graphical output

CLOSEWIND Set the output back to screen output if previous output has been to a .ps file.

SETPAR Set some parameters mainly colours and number of contours.

SECX Vertical intersection, y-z plot.

SECTIONX Vertical intersection, y-z plot.

SECY Vertical intersection, x-z plot.

SECTIONY Vertical intersection, x-z plot.

SECZ Horizontal intersection with wind arrows, x-y plot.

SECTIONZ Horizontal intersection with wind arrows, x-y plot.

VECTPLOTS2D Arrow plot routine.

COMPCOLOR Linear interpolation in colours.

ER Make error plots.

Bibliography

[1] Cullen, M.J.P., Norbury, J., Purser, R.J. & Shutts, G.J. 1987 Modelling the quasi-geostrophics dynamics of the atmosphere. *Quart. J. R. Met. Soc.* **113**(2), 735-757

[2] Fulton, S.R. 1989 Multigrid solution of the semigeostrophic invertibility relation. *Monthly Weather Review* **117**, 2059-2066.

[3] Hoskins, B.J., McIntyre, M.E. & Robertson, A.W. 1985 On the use and significance of isentropic potential vorticity maps. *Quart. J. R. Met. Soc.* **111**, 877-946.

[4] Ortega, J.M. & Rheinboldt, W.C. 1970 *Iterative solution of nonlinear equations in several variables.* Academic Press, New York and London.

[5] Pogorelov, A.V. 1964 *Monge Ampere equations of the elliptic type* . Noordhoff Groningen, the Netherlands.

[6] Roulstone, I. & Norbury J. 1994 A Hamiltonian structure with contact geometry for the semi-geostrophic equations. *J. Fluid Mech.* **272** , 211-233.

[7] Stüben, K. & Trottenberg, U. 1981 *Multigrid methods: Fundamental algorithms, model problem analysis & applications.* Lecture notes in mathematics 960, Springer-Verlag Berlin, 1-176.

[8] Thorpe, A.J. 1985 Diagnosis of balanced vortex structure using potential vorticity, *J. Atmos. Sci.* **42**(4), 1384-1389.

[9] Thorpe, A.J. 1986 Synoptic scale disturbances with circular symmetry, *Mon Wea. Rev.* **114**, 1384-1389.